

Combining Multiple Clusterings via Crowd Agreement Estimation and Multi-Granularity Link Analysis

Dong Huang^a, Jian-Huang Lai^{a,*}, Chang-Dong Wang^{b,c}

^a*School of Information Science and Technology, Sun Yat-sen University, P.R. China*

^b*School of Mobile Information Engineering, Sun Yat-sen University, P.R. China*

^c*SYSU-CMU Shunde International Joint Research Institute (JRI), P.R. China*

Abstract

The clustering ensemble technique aims to combine multiple clusterings into a probably better and more robust clustering and has been receiving an increasing attention in recent years. There are mainly two aspects of limitations in the existing clustering ensemble approaches. Firstly, many approaches lack the ability to weight the base clusterings without access to the original data and can be affected significantly by the low-quality, or even ill clusterings. Secondly, they generally focus on the instance level or cluster level in the ensemble system and fail to integrate multi-granularity cues into a unified model. To address these two limitations, this paper proposes to solve the clustering ensemble problem via crowd agreement estimation and multi-granularity link analysis. We present the normalized crowd agreement index (NCAI) to evaluate the quality of base clusterings in an unsupervised manner and thus weight the base clusterings in accordance with their clustering validity. To explore the relationship between clusters, the source aware connected triple (SACT) similarity is introduced with regard to their common neighbors and the source reliability. Based on NCAI and multi-granularity information collected among base clusterings, clusters, and data instances, we further

[☆]Requests for the code should be sent to the first author via email.

^{*}Corresponding author. Present address: School of Information Science and Technology, Sun Yat-sen University, Guangzhou Higher Education Mega Center, Panyu District, Guangzhou, Guangdong, 510006, P. R. China. Tel.: +86-13168313819. Fax: +86-20-84110175.

Email addresses: huangdonghere@gmail.com (Dong Huang), stsljh@mail.sysu.edu.cn (Jian-Huang Lai), changdongwang@hotmail.com (Chang-Dong Wang)

propose two novel consensus functions, termed weighted evidence accumulation clustering (WEAC) and graph partitioning with multi-granularity link analysis (GP-MGLA) respectively. The experiments are conducted on eight real-world datasets. The experimental results demonstrate the effectiveness and robustness of the proposed methods.

Keywords: Weighted clustering ensemble, Weighted consensus clustering, Weighted evidence accumulation clustering, Graph partitioning with multi-granularity link analysis

1. Introduction

Data clustering is a fundamental and very challenging problem in data mining and machine learning. The purpose is to partition unlabeled data into homogeneous groups, each referred to as a cluster. Data clustering requires a distance metric for evaluating the similarity between data instances, which, without prior knowledge of cluster shapes, is hard to specify. In the past few decades, a large number of clustering algorithms have been developed [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. However, there is no single clustering method which is able to identify all sorts of cluster shapes and structures in data.

For the same dataset, different methods, or even the same method with different initializations or parameter settings, may lead to very different clustering results. It is extremely difficult to decide which method would be the *proper* one for a given clustering task, not to say how to properly specify the initialization and parameter setting for the chosen method. Each method has its own merits as well as weaknesses. Different clusterings generated by different methods or with varying parameters can provide multiple views of the data. How to combine the information of different clustering results for obtaining a better and more robust clustering remains a very challenging problem [11, 12].

In recent years, many clustering ensemble approaches have been developed, which aim to combine multiple clusterings into a probably better and more robust clustering by utilizing various techniques [13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25]. However, in most of the existing methods, there are mainly two aspects of limitations. Firstly, many of the clustering ensemble approaches lack the ability to weight the base clusterings without access to the original data features, which makes them vulnerable to low-quality clusterings and probable to be affected significantly by low-quality clusterings

(or even ill clusterings). Secondly, they mainly focus on the instance level or the cluster level in the ensemble system and fail to fuse multi-granularity information into a unified model. In order to address these two limitations, in this paper, we propose a clustering ensemble framework based on crowd agreement estimation and multi-granularity link analysis. By exploring the relationship among the base clusterings, we present a novel clustering validity measure termed normalized crowd agreement index (NCAI), which is able to evaluate the quality of base clusterings in an unsupervised manner and provides information for treating each base clustering accordingly. The source aware connected triple (SACT) similarity is introduced for analyzing the similarity between clusters with regard to their common neighbors and source reliability. Besides the relations between base clusterings and between clusters, we further investigate the linkage between data instances and clusters and incorporate the information from the three levels of granularity in a unified framework. In our previous work [26], we introduced the consensus function termed graph partitioning with multi-granularity link analysis (GP-MGLA). This paper is a major extension of our previous work on clustering ensemble. In this paper, more comprehensive literature and motivation are provided. Besides that, we propose another novel consensus function termed weak evidence accumulation clustering (WEAC), which is developed from the conventional evidence accumulation clustering (EAC) [15] and capable of dealing with ill clusterings by incorporating the clustering validity cue into the ensemble process. Extensive experiments are further conducted on real-world datasets for evaluating the proposed methods against several baseline clustering ensemble methods.

The remainder of this paper is organized as follows. In Section 2, we review the related work of the clustering ensemble technique. In Section 3, we describe the formulation of the clustering ensemble problem. In Section 4, we present the crowd agreement estimation mechanism. The source aware connected triple (SACT) similarity is introduced in Section 5. In Section 6, we propose two novel consensus functions termed weighted evidence accumulation clustering (WEAC) and graph partitioning with multi-granularity link analysis (GP-MGLA) respectively. The experimental results are reported in Section 7. We conclude this paper in Section 8.

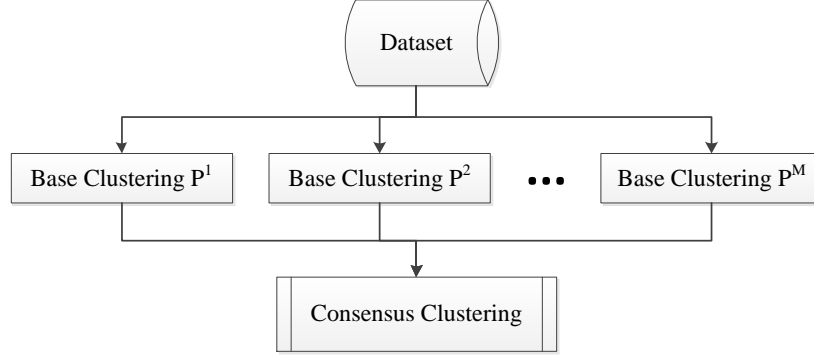


Figure 1: The clustering ensemble process.

2. Related Work

Clustering ensemble is also known as clustering combination or clustering aggregation, which aims to combine multiple clusterings, each referred to as a base clustering (or an ensemble member), to obtain a so-called consensus clustering. As illustrated in Fig. 1, the clustering ensemble process involves two steps: the first step is to generate multiple clusterings for a given dataset; and the second step is to construct the consensus clustering from the ensemble of base clusterings using different consensus functions.

Given a dataset, the ensemble of base clusterings can be generated by running different clustering algorithms [22, 24, 26], running the same algorithm with different initializations and parameters [15, 19, 21, 23], clustering via sub-sampling the data repeatedly [13, 14], or clustering via projecting the data onto different subspaces [13, 14, 16, 20]. Compared to generating base clusterings, how to combine multiple base clusterings, i.e., how to design the consensus function, is much more important and challenging in the clustering ensemble problem.

In the past few years, many consensus functions have been developed to fuse information from multiple clusterings [13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25]. These approaches can be classified into mainly three categories, namely, (i) the median partition based methods [27, 16, 25], (ii) the pair-wise co-occurrence based methods [15, 18, 21], and (iii) the graph partitioning based methods [13, 14, 20].

In the median partition based approaches [27, 16, 25], the clustering ensemble problem is formulated into an optimization problem, aiming to find

the partition/clustering that maximizes the similarity between the the partition and the base clusterings, over the space of all partitions. The median partition problem is NP-complete [16]. Instead of finding the optimal solution over the huge space of all possible partitions, Cristofor and Simovici [27] used the genetic algorithm to obtain an approximative solution where the clusterings are represented by chromosomes. Topchy et al. [16] cast the median partition problem into a maximum likelihood problem, as a solution to which the consensus clustering is found using the EM algorithm. Franek and Jiang [25] reduced the median partition problem to the Euclidean median problem by clustering embedding in vector spaces and found the median vector by the Weiszfeld algorithm [28]. Then an inverse transformation would be performed to convert the median vector into a clustering, which was taken as the consensus clustering.

The pair-wise co-occurrence based approaches [15, 18, 21] construct the similarity between data instances by considering how many times they occur in the same cluster in the ensemble of base clusterings. Fred and Jain [15] introduced the evidence accumulation clustering (EAC) method, which used the co-association matrix to measure the similarity between instances. Then the hierarchical agglomerative clustering algorithms [11], e.g., single-link (SL) and average-link (AL), can be performed on the co-association matrix and thus the consensus clustering is obtained. Li et al. [18] analyzed the co-association matrix and proposed a novel hierarchical clustering algorithm by utilizing the concept of normalized edges to measure the similarity between clusters. Wang et al. [21] generalized the EAC method and proposed the probability accumulation method, which took into consideration the sizes of clusters in the ensemble.

Another category of clustering ensemble is based on graph partitioning [13, 14, 20]. Strehl and Ghosh [13] modeled the ensemble of clusterings in a hypergraph structure where the clusters are treated as hyperedges. For partitioning the graph and obtaining the consensus clustering, they further proposed three graph partitioning algorithms, namely, the cluster-based similarity partitioning algorithm (CSPA), the hypergraph-partitioning algorithm (HGPA), and the meta-clustering algorithm (MCLA). Fern and Brodley [14] formulated the clustering ensemble into a bipartite graph where both the data instances and clusters are represented as graph nodes. An edge between two nodes exists if and only if one of the nodes is a data instance and the other node is the cluster containing it. The consensus clustering is obtained by partitioning the graph into a certain number of disjoint sets of

graph nodes.

Many of the existing clustering ensemble approaches implicitly assume that all the base clusterings contribute equally to the ensemble system and can be affected significantly by low-quality clusterings or even ill clusterings. In recent years, some efforts have been made to weight the base clusterings with regard to the clustering validity. Vega-Pons et al. [29] exploited several property validity indexes (PVI), namely, Variance (VI), Connectivity (CI), Silhouette Width (SI) and Dunn index (DI), to assign a weight to each partition in the ensemble and proposed a new clustering ensemble method based on kernel functions. Vega-Pons et al. [30] also extended the conventional EAC method by weighting the partitions based on the PVI. These PVI need access to the original feature vectors, which are not supposed to be given for the consensus process in the formulation of this work as well as many other clustering ensemble frameworks [14, 15, 18, 19, 21, 22, 23, 24, 25]. Li and Ding [31] proposed the weighted consensus clustering (WCC) method, where the weights of the base clusterings are determined via an optimization process based on the nonnegative matrix factorization. The optimization process is computationally expensive when dealing with large datasets. Fern and Lin [32] proposed a clustering ensemble selection framework which selects a subset of partitions from a large library of partitions. The ensemble selection process in [32] can be viewed as weighting the partitions in the ensemble with either 1 or 0, where 1's indicate the preserved partitions and 0's indicate the deleted ones. However, the ensemble selection scheme lacks the flexibility of weighting the selected members in accordance to their quality.

3. The Clustering Ensemble Problem

The purpose of a clustering algorithm is to discover the structure of clusters in a given dataset. The clustering result can be either a hard partition or a fuzzy partition for the dataset. The clustering ensemble technique aims to combine multiple partitions for achieving a better partition. In this paper, we focus on combining hard partitions of data.

Given a dataset $\mathcal{X} = \{x_1, x_2, \dots, x_n\}$, where x_i is the i -th data instance and n is the number of instances in \mathcal{X} . A partition (or clustering) of \mathcal{X} is generated by running a clustering algorithm with some specific parameters. Each cluster in a partition consists a certain number of data instances. Different clusters in the same partition do not intersect with each other. And the union of all clusters in a partition covers the entire dataset.

Formally, let

$$P^i = \{C_1^i, C_2^i, \dots, C_{n_i}^i\} \quad (1)$$

be a partition of \mathcal{X} , where C_j^i denotes the j -th cluster and n_i is number of clusters in P^i . Then we have $\forall C_j^i \in P_i, C_j^i \neq \emptyset, \forall j \neq k, C_j^i \cap C_k^i = \emptyset$, and $\bigcup_{j=1}^{n_i} C_j^i = P^i$.

In a clustering ensemble system, each partition is referred to as a base clustering. With the partitions generated by different algorithms or the same algorithm with different parameters and initializations, we can obtain the ensemble of M base clusterings, which is denoted as

$$\mathcal{P} = \{P^1, P^2, \dots, P^M\}, \quad (2)$$

where P^i represents the i -th base clustering in \mathcal{P} . For convenience, the set of all clusters in the ensemble is denoted as $\mathcal{C} = \{C_1, C_2, \dots, C_{n_c}\}$, where C_i is the i -th cluster in \mathcal{C} . As is defined, it holds that $\mathcal{C} = \bigcup_{i=1}^M P^i$ and $n_c = \sum_{i=1}^M n_i$.

The multiple partitions of \mathcal{X} provide multiple looks at the dataset. The problem is to use the information provided by the the ensemble of multiple partitions to obtain a final partition solution P^* , which is generally referred to as the consensus clustering.

4. Crowd Agreement Estimation

In the clustering ensemble system, the base clusterings can be generated using a wide variety of clustering algorithms. Due to the diversity of clustering algorithms and datasets, it is not guaranteed that every base clustering is well constructed. The low-quality clusterings, or even ill clusterings, may affect the consensus process significantly. There is a need to distinguish the poor clusterings from the good ones and treat the base clusterings with regard to their quality. The critical problem here is how to evaluate the quality of the base clusterings without knowing the ground-truth.

Some algorithms have been developed to estimate the clustering quality using different criteria [33, 34, 35]. Wu and Chow [33] proposed a clustering validity index based on inter-cluster and intra-cluster density. Faceli et al. [34] used the overall deviation and the connectivity to assess the quality of a clustering. The overall deviation of a clustering measures the overall distances between data instances and their corresponding cluster centers. The connectivity measures how often neighboring instances are assigned to the

same cluster. Li and Latecki [35] utilized the average silhouette coefficient to evaluate the quality of a cluster. The silhouette coefficient of a data instance measures how similar that instance is to the instances in its own cluster compared to the instances in the other clusters, whereas the quality of a cluster is estimated by the average of the silhouette coefficients of the instances inside it. These evaluation methods are only applicable to numerical data and need access to the original data features, which are not supposed to be given in the problem formulation of many clustering ensemble approaches [14, 15, 18, 19, 21, 22, 23, 24, 25]. Rather than utilizing the information of data distribution, in this paper, we view the clustering ensemble as a crowd of individuals and estimate the quality of each individual via consulting the other individuals in the clustering ensemble.

In social and economic science, “the wisdom of the crowd” is the process of taking into consideration the collective opinion of a crowd of individuals rather than a single expert [36]. The ground-truth labeling of a dataset can be viewed as an expert. As the ground-truth is not supposed to be known in unsupervised frameworks, we estimate the quality of a base clustering by collecting information from the crowd of base clusterings. Each base clustering is compared with the other ones and the average opinion of the crowd of individuals is obtained for quality estimation.

Definition 1. Let \mathcal{P} be an ensemble of base clusterings and P^i be the i -th base clustering in \mathcal{P} . The crowd agreement index (CAI) for P^i is defined as

$$CAI(P^i) = \frac{1}{M-1} \sum_{P^j \in \mathcal{P}, i \neq j} Sim(P^i, P^j), \quad (3)$$

where $Sim(P^i, P^j)$ denotes the similarity between the two base clusterings P^i and P^j .

We denote the base clustering that gains the maximum agreement from the crowd as the reference member. Then the reliability of the base clusterings is estimated by comparing their crowd agreement with that of the reference member and the normalized version of crowd agreement index can be computed.

Definition 2. The normalized crowd agreement index of P^i is defined as

$$NCAI(P^i) = \frac{CAI(P^i)}{\max_{P^j \in \mathcal{P}} CAI(P^j)}. \quad (4)$$

The basic idea here is to estimate the quality of a base clustering by collecting opinion from a crowd of diverse individuals. According to Definition 2, for $i = 1, 2, \dots, M$, it holds that $NCAI(P^i) \in [0, 1]$. In this paper we use the normalized mutual information (NMI) [13] as the similarity measure $Sim(P^i, P^j)$. The greater the NCAI value of a base clustering is, the better its quality is supposed to be.

5. Source Aware Connected Triple

In this section, we investigate the relationship among the clusters in the ensemble and introduce the source aware connected triple (SACT) which is able to measure the similarity of two clusters with regard to their common neighbors and the source reliability.

Definition 3. *Two clusters C_i and C_j are neighbors if and only if they share some common data instances, i.e., $C_i \cap C_j \neq \emptyset$.*

Each cluster is a set of data instances. The Jaccard coefficient [37] is often used to measure the similarity between two clusters (or two sets), which is computed as follows:

$$J(C_i, C_j) = \frac{|C_i \cap C_j|}{|C_i \cup C_j|}, \quad (5)$$

where C_i and C_j are two clusters and $|S|$ denotes the cardinality of the set S . The Jaccard coefficient takes into consideration the sharing instances of two clusters to measure their similarity. Therefore the Jaccard coefficient of two clusters in the same base clustering is always zero. If two clusters intersect, then they are directly related. If two clusters do not intersect but they share a certain number of common neighbors, then they are also related. Iam-On et al. [23] utilized the information of common neighbors of two clusters to justify their similarity, where, however, the reliability of these neighbors was not considered.

Each base clustering can be viewed as a source of clusters. The overall quality of the clusters in a base clustering is correlated to the quality of the base clustering containing them. In this paper, we estimate the reliability of a cluster by considering the quality of the corresponding base clustering and propose the source aware connected triple (SACT) to measure the similarity of two clusters with regard to their common neighbors and the reliability of these neighbors.

Definition 4. The SACT coefficient between two clusters C_i and C_j w.r.t. a cluster C_k is defined as

$$SACT_{ij}^k = I_{NCAI}(P(C_k)) \cdot \min(J(C_i, C_k), J(C_j, C_k)), \quad (6)$$

where $P(C_k)$ denotes the base clustering that contains C_k and

$$I_{NCAI}(P^l) = (NCAI(P^l))^\beta \quad (7)$$

is the influence of the NCAI of the base clustering P^l .

According to Definitions 2 and 4, for $l = 1, 2, \dots, M$, it holds that $I_{NCAI}(P^l) \in [0, 1]$. The parameter $\beta > 0$ in Eq. (7) is a parameter to adjust the influence of the NCAI. A greater value of β leads to a bigger influence of the NCAI, which means the difference of NCAI values between high-confidence partitions and low-confidence partitions is enlarged. When $\beta = 0$, the influence of NCAI disappears for all base clusterings, i.e., $\forall P^l \in \mathcal{P}, I_{NCAI}(P^l) = 0$.

Definition 5. The SACT coefficient between two clusters C_i and C_j w.r.t. all the clusters in the ensemble \mathcal{C} is defined as

$$SACT_{ij} = \sum_{C_k \in \mathcal{C}} SACT_{ij}^k. \quad (8)$$

By definition, if C_k is not a common neighbor between C_i and C_j , then $SACT_{ij}^k = 0$. Thus the SACT coefficient between two clusters w.r.t. all the common neighbors is identical to that w.r.t. all the clusters in the ensemble and can be computed by Eq. (8).

Definition 6. The SACT similarity between two clusters C_i and C_j is defined as

$$SIM_{SACT}(C_i, C_j) = \begin{cases} 1, & \text{if } i = j, \\ \frac{SACT_{ij}}{\max_{\forall C_x, C_y \in \mathcal{C}} SACT_{xy}}, & \text{otherwise.} \end{cases} \quad (9)$$

The SACT similarity is computed on the basis of the the SACT coefficient. The pair of clusters with the maximum SACT coefficient is adopted as the reference pair of clusters, whose SACT similarity is defined to be 1. The SACT similarity of the other pairs of clusters is computed by comparing their SACT coefficient to that of the reference pair (see Eq. (9)). The SACT similarity between a cluster and itself is set to 1.

6. Consensus Functions

In this section, we introduce two novel consensus functions which utilize multi-granularity information of the ensemble and are able to deal with ill base clusterings. In the following, we will describe the weighted evidence accumulation clustering (WEAC) method in Section 6.1 and the graph partitioning with multi-granularity link analysis (GP-MGLA) method in Section 6.2.

6.1. Weighted Evidence Accumulation Clustering (WEAC)

In a base clustering, each data instance is assigned to a specific cluster, whereas two instances are either in the same cluster or in two different clusters. Without access to the original features, the affinity between two data instances can be assessed by their co-occurrence information in the ensemble of base clusterings.

Definition 7. Let P^l be a base clustering in the clustering ensemble \mathcal{P} . Let $P^l(i)$ be the cluster label of the instance i in P^l . The $n \times n$ similarity matrix S^l for P^l is computed as follows:

$$S_{ij}^l = \begin{cases} 1, & \text{if } P^l(i) = P^l(j), \\ 0, & \text{otherwise,} \end{cases} \quad (10)$$

for $i = 1, \dots, n, j = 1, \dots, n$.

For each base clustering, say, P^l , a similarity matrix S^l is constructed. If instances i and j occur in the same cluster in P^l , then $S_{ij}^l = 1$; otherwise $S_{ij}^l = 0$. The similarity matrix contains the pair-wise co-occurrence information of the corresponding base clustering. In the conventional evidence accumulation clustering (EAC) method [15], the association matrix A is obtained by averaging the similarity matrices of all the base clustering, that is

$$A = \frac{1}{M} \sum_{l=1}^M S^l. \quad (11)$$

The basic idea of the proposed WEAC method is to construct the association matrix with considering the reliability of the base clusterings. We assess the quality of each base clustering with the NCAI measure (as described in Section 4) and assign a weight to each base clustering with regard to its estimated quality.

Definition 8. The weighted co-association matrix \tilde{A} is a $n \times n$ matrix which is computed as follows:

$$\tilde{A} = \sum_{l=1}^M w_l S^l, \quad (12)$$

where

$$w_l = \frac{I_{NCAI}(P^l)}{\sum_{i=1}^M I_{NCAI}(P^i)} \quad (13)$$

is the weight of the base clustering P^l .

According to Definitions 7 and 8, for $i = 1, \dots, n$ and $j = 1, \dots, n$, it holds that $\tilde{A}_{ij} \in [0, 1]$. Thus the labeling information of multiple base clusterings is mapped into a new similarity measure by utilizing pair-wise co-occurrence cues and reliability assessment of each member. With the weighted co-association matrix constructed, we further perform the agglomerative clustering methods [11] to achieve the final consensus clustering.

For clarity, the WEAC method is summarized in Algorithm 1.

Algorithm 1 (Weighted Evidence Accumulation Clustering)

Input: \mathcal{P} , k .

- 1: Initialization:
Evaluate the quality of each base clustering in \mathcal{P} with NCAI according to Eq. (4) and (7).
- 2: **for** $l = 1, 2, \dots, M$ **do**
- 3: Construct the similarity matrix M^l for P^l according to Eq. (10).
- 4: **end for**
- 5: Build the weighted co-association matrix \tilde{A} according to Eq. (12).
- 6: Use the agglomerative methods to obtain the consensus clustering with k clusters.

Output: the consensus clustering P^* .

6.2. Graph Partitioning with Multi-Granularity Link Analysis (GP-MGLA)

There are three levels of granularity in the clustering ensemble, namely, the data instances, the clusters, and the base clusterings. The existing methods mainly focus on the level of data instances and that of clusters and lack the ability to treat the three levels of granularity as a whole system. In this section, we proposed a graph based clustering ensemble method termed graph partitioning with multi-granularity link analysis (GP-MGLA). In the

proposed GP-MGLA method, we formulate the three levels of granularity in the clustering ensemble into a bipartite graph model, which will be described in the following.

Compared to the previous clustering ensemble methods based on graph partitioning [13, 14], the GP-MGLA method is distinguished mainly in two aspects. Firstly, the GP-MGLA method utilizes the crowd agreement estimation mechanism (see Section 4) for exploiting the relationship among base clusterings and evaluating the quality of the base clusterings in an unsupervised manner. Secondly, the links between clusters are integrated into the graph model via the SACT similarity measure.

In our bipartite graph model, both data instances and clusters are treated as graph nodes. There are two types of links in the graph, that is, the links between instances and the cluster containing them and the links between clusters that have common neighbors. To implement the bipartite structure, each cluster is used twice, i.e., for each cluster, there are two different nodes representing it in the bipartite graph.

Formally, we construct the bipartite graph as follows:

$$G = (U, V, L), \quad (14)$$

where $U = \mathcal{X} \cup \mathcal{C}$ is the set of nodes including all instances and clusters, $V = \mathcal{C}$ is the set of nodes including all clusters, and L is the set of graph links. The graph G is an undirected graph. There are no links between the nodes in U or between the nodes in V . All links are constructed between the nodes in U and those in V .

Let $u_i \in U$ and $v_j \in V$ be two nodes in the graph G . If u_i is a data instance and v_j is the cluster containing u_i , then a link exists between u_i and v_j and the link between them is weighted with regard to the quality of the base clustering that v_j belongs to. If both u_i and v_j are clusters, then the link between them is constructed via the SACT measure (see Section 5). Formally, the weight of the link between the nodes u_i and v_j is defined as follows:

$$w_{ij} = \begin{cases} \alpha \cdot I_{NCAI}(P(v_j)), & \text{if } u_i \in \mathcal{X}, v_j \in \mathcal{C}, u_i \in v_j, \\ SIM_{SACT}(u_i, v_j), & \text{if } u_i \in \mathcal{C}, v_j \in \mathcal{C}, \\ 0, & \text{otherwise.} \end{cases} \quad (15)$$

In the graph G , the instances and the clusters are used as nodes and the relationship among them is incorporated into the graph links. Also the

information among the base clusterings is exploited to provide a reliability measure for the graph links via the crowd agreement estimation. With regard to the bipartite structure of the graph G , the Tcut algorithm [38] can be utilized for partitioning the graph into a specific number of disjoint sets of nodes. The data instances in each of these disjoint sets are treated as a cluster and thus the final consensus clustering is obtained. Theoretically, there is a possibility that some of these disjoint sets consist of only clusters and no instances, which would lead to a less number of clusters than specified. However, we have never come across this situation in our experiments, probably due to that the joint force of the links between the instances and clusters containing them is strong enough to hold at least part of them together. For clarity, we summarize the GP-MGLA method in Algorithm 2.

Algorithm 2 (Graph Partitioning with Multi-Granularity Link Analysis)

Input: \mathcal{P} , k .

- 1: Initialization:
 Evaluate the quality of each base clustering in \mathcal{P} with NCAI according to Eq. (4) and (7).
 Compute the SACT similarity between clusters according to Eq. (8).
- 2: Build the bipartite graph $G = (U, V, L)$ with $U = \mathcal{X} \cup \mathcal{C}$, $V = \mathcal{C}$, and L constructed as Eq. (15).
- 3: Partition the graph G with the Tcut algorithm into k disjoint sets of nodes.
- 4: Treat the data instances in each set as a cluster and thus obtain the consensus clustering.

Output: the consensus clustering P^* .

7. Experiments

In this section, we conduct experiments on eight real-world datasets and compare the proposed approaches against several baseline clustering ensemble approaches. The datasets and evaluation criterion are described in Section 7.1. The setting of parameters is discussed in Sections 7.2. The construction of base clusterings is introduced in Section 7.3. Then we evaluate the performance of the proposed methods compared to the baseline methods in Section 7.4. The analysis of computational complexity is presented in Sections 7.5.

The experiments in this paper are conducted in Matlab 7.14.0.739 (R2012a) 64-bit on a workstation (Windows Server 2008 R2 64-bit, 8 Intel 2.40GHz processors, 96GB of RAM).

7.1. Datasets and Evaluation Criterion

Table 1: Description of the benchmark datasets

| Dataset | #Instance | #Attribute | #Class |
|---------------------------|-----------|------------|--------|
| <i>Breast Cancer</i> | 683 | 9 | 2 |
| <i>Image Segmentation</i> | 2,310 | 19 | 7 |
| <i>Iris</i> | 150 | 4 | 3 |
| <i>Seeds</i> | 210 | 7 | 3 |
| <i>Yeast</i> | 1,484 | 8 | 10 |
| <i>Wine</i> | 178 | 13 | 3 |
| <i>Pen Digits</i> | 10,992 | 16 | 10 |
| <i>Letters</i> | 20,000 | 16 | 26 |

In our experiments, eight real-world datasets from the UCI machine learning repository [39] are used, namely, *Breast Cancer*, *Image Segmentation*, *Iris*, *Seeds*, *Yeast*, *Wine*, *Pen Digits*, and *Letters*. The details of the benchmark datasets are given in Table 1.

To evaluate the quality of the consensus clustering, we utilize the normalized mutual information (NMI) [13] which provides an indication of the shared information between two clusterings. Let P^* be the test clustering and P^G the ground-truth clustering. The NMI score of P^* w.r.t. P^G is computed as follows:

$$NMI(P^*, P^G) = \frac{\sum_{i=1}^{n^*} \sum_{j=1}^{n^G} n_{ij} \log \frac{n_{ij}n}{n_i^* n_j^G}}{\sqrt{\sum_{i=1}^{n^*} n_i^* \log \frac{n_i^*}{n} \sum_{j=1}^{n^G} n_j^G \log \frac{n_j^G}{n}}}, \quad (16)$$

where n^* is the number of clusters in P^* , n^G is the number of clusters in P^G , n_i^* is the number of instances in the i -th cluster of P^* , n_j^G is the number of instances in the j -th cluster of P^G , and n_{ij} is the number of common instances shared by cluster i in P^* and cluster j in P^G .

Table 2: The performance of WEAC with varying parameters in terms of NMI

| Dataset | β | | | | |
|---------------------------|---------|-------|-------|-------|-------|
| | 0 | 1 | 2 | 4 | 8 |
| <i>Breast Cancer</i> | 0.647 | 0.673 | 0.674 | 0.687 | 0.685 |
| <i>Iris</i> | 0.734 | 0.743 | 0.778 | 0.748 | 0.750 |
| <i>Image Segmentation</i> | 0.639 | 0.641 | 0.648 | 0.647 | 0.657 |
| <i>Seeds</i> | 0.591 | 0.623 | 0.634 | 0.624 | 0.626 |
| <i>Yeast</i> | 0.230 | 0.234 | 0.232 | 0.241 | 0.239 |
| <i>Wine</i> | 0.753 | 0.772 | 0.781 | 0.781 | 0.757 |
| <i>Pen Digits</i> | 0.742 | 0.753 | 0.770 | 0.777 | 0.796 |
| <i>Letters</i> | 0.434 | 0.443 | 0.444 | 0.451 | 0.454 |

Table 3: The performance of GP-MGLA with varying parameters in terms of NMI

| α | 0.5 | | | | 0.01 | 0.1 | 1 |
|---------------------------|-------|-------|-------|-------|-------|-------|-------|
| β | 0 | 2 | 4 | 8 | 2 | | |
| <i>Breast Cancer</i> | 0.677 | 0.719 | 0.725 | 0.729 | 0.702 | 0.712 | 0.713 |
| <i>Iris</i> | 0.739 | 0.742 | 0.742 | 0.751 | 0.743 | 0.748 | 0.742 |
| <i>Image Segmentation</i> | 0.635 | 0.650 | 0.648 | 0.649 | 0.639 | 0.642 | 0.651 |
| <i>Seeds</i> | 0.593 | 0.620 | 0.621 | 0.609 | 0.611 | 0.614 | 0.623 |
| <i>Yeast</i> | 0.239 | 0.251 | 0.252 | 0.250 | 0.243 | 0.246 | 0.249 |
| <i>Wine</i> | 0.781 | 0.798 | 0.792 | 0.783 | 0.788 | 0.786 | 0.794 |
| <i>Pen Digits</i> | 0.779 | 0.796 | 0.803 | 0.800 | 0.788 | 0.792 | 0.798 |
| <i>Letters</i> | 0.448 | 0.456 | 0.456 | 0.461 | 0.449 | 0.454 | 0.456 |

7.2. Choices of Parameters

There is one parameter β in the WEAC method and two parameters α and β in the GP-MGLA method. The parameter α is a scale factor for the link weights between instances and clusters. The parameter β adjusts the influence of NCAI for both WEAC and GP-MGLA, where a bigger β signals a greater influence of NCAI. We evaluate the performance of the proposed WEAC and GP-MGLA methods with varying parameters on the benchmark datasets. As can be seen in Table 2 and 3, the proposed methods are very stable w.r.t. the varying parameters. Empirically, it is suggested that α be set in the interval of (0.1, 1) and β be set in the interval of (1, 4) for the proposed two methods. In the following, the parameters are set that $\alpha = 0.5$ and $\beta = 2$ for all the experiments on all the benchmark datasets.

7.3. Generation of Base Clustering Ensemble

The proposed approaches make no specific assumption about the generation of the ensemble of base clusterings. To evaluate the effectiveness and robustness of the proposed methods over various combinations of base clusterings, we construct a pool of a large number of different base clusterings. Then we run the proposed methods and the baseline methods with the base clusterings randomly chosen from the pool repeatedly.

Four clustering algorithms are used to construct the base clustering pool, namely, k -means, rival penalized competitive learning (RPCL) [1], hierarchical mode association clustering (HMAC) [2], and incremental support vector clustering with outlier detection (OD-ISVC) [40]. To obtain a pool of various base clusterings, we apply the aforementioned clustering algorithms repeatedly with random parameters and initializations on each dataset. The number of clusters for the k -means and RPCL methods are randomly chosen in the interval of $[2, 2\sqrt{n}]$, where n is the number of instances in the dataset. The HMAC method is a hierarchical clustering method. We choose the hierarchy of clustering randomly for the HMAC method, where each hierarchy corresponds to a clustering with a certain number of clusters. For the OD-ISVC method, the base clusterings are generated with randomly chosen kernel width parameter q and trade-off parameter C . In this paper, we apply each of the clustering algorithms for 100 times and thus a pool of 400 different base clusterings is constructed for each dataset.

7.4. Performance Comparison and Analysis

With the base clustering pool constructed (see Section 7.3), the proposed approaches and the baseline approaches are applied to the ensemble of base clusterings which are randomly chosen from the pool. In our experiments, each of the clustering ensemble approaches has no knowledge about how the chosen base clusterings are generated, i.e., by which algorithm and with what parameters they are generated. For each run, an ensemble of M base clusterings is randomly constructed and different clustering ensemble approaches are applied to the ensemble. The ensemble size $M = 5$ is used in our work. We test the proposed approaches against the baseline approaches by evaluating their performance over a large number of runs, which aims to rule out the factor of “*getting lucky sometimes*” and provide a fair comparison for their effectiveness and robustness over different combinations of base clusterings.

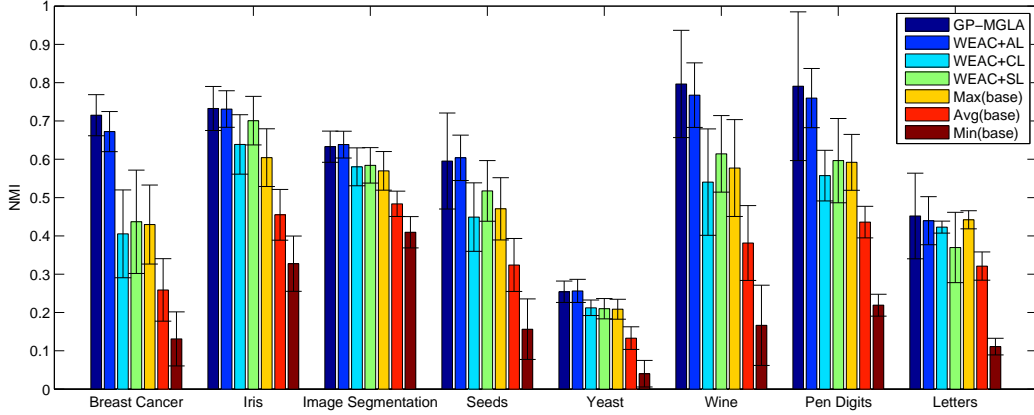


Figure 2: Average performance in terms of NMI over 100 runs by WEAC and GP-MGLA compared to the base clusterings.

7.4.1. Comparison with Base Clusterings

In this paper we propose two novel consensus functions, namely, the GP-MGLA method and the WEAC method. GP-MGLA is a graph partitioning based method, whereas WEAC is a pair-wise similarity based method. With the weighted co-association matrix computed by WEAC, we further perform three agglomerative methods, namely, average-link (AL), complete-link (CL), and single-link (SL) to obtain the final consensus clustering, which leads to three sub-methods denoted as WEAC-AL, WEAC-CL, and WEAC-SL respectively.

For each run, an ensemble is generated by randomly drawing M base clusterings from the pool. We apply the proposed clustering ensemble methods on different ensembles for each dataset repeatedly. The average performance over 100 runs of our methods compared to the base clusterings is shown in Fig. 2, in which Max(base) denotes the average NMI score of the best base clustering over all ensembles, Min(base) denotes the average NMI score of the worst base clustering over all ensembles, and Avg(base) denotes the average NMI score of all base clusterings over all ensembles. As shown in Fig. 2, the proposed methods are able to produce better and more robust consensus clusterings than the base clusterings. Specially, GP-MGLA and WEAC-AL significantly outperform the base clusterings on the *Breast Cancer*, *Seeds*, *Wine*, and *Pen Digits* datasets.

We further compare the consensus clusterings by our methods against

Table 4: Winning percentage of the consensus clustering against base clusterings w.r.t. the best number of clusters over 100 runs.

| Dataset | <i>Breast Cancer</i> | <i>Image Segmentation</i> | <i>Iris</i> | <i>Seeds</i> |
|---------|----------------------|---------------------------|-------------------|----------------|
| GP-MGLA | 99.6% | 97.2% | 98.4% | 99.2% |
| WEAC | 99.0% | 99.2% | 98.4% | 99.2% |
| Dataset | <i>Yeast</i> | <i>Wine</i> | <i>Pen Digits</i> | <i>Letters</i> |
| GP-MGLA | 100% | 98.0% | 100% | 99.0% |
| WEAC | 100% | 96.8% | 100% | 70.6% |

Table 5: Winning percentage of the consensus clustering against base clusterings w.r.t. the same number of clusters over 100 runs.

| Dataset | <i>Breast Cancer</i> | <i>Image Segmentation</i> | <i>Iris</i> | <i>Seeds</i> |
|---------|----------------------|---------------------------|-------------------|----------------|
| GP-MGLA | 59.7% | 72.5% | 70.7% | 65.8% |
| WEAC | 67.4% | 76.0% | 70.0% | 71.1% |
| Dataset | <i>Yeast</i> | <i>Wine</i> | <i>Pen Digits</i> | <i>Letters</i> |
| GP-MGLA | 64.5% | 68.1% | 95.7% | 73.0% |
| WEAC | 68.1% | 73.8% | 97.3% | 66.6% |

each of the base clusterings and calculate the winning percentage. For each run, an ensemble of M base clusterings are selected. Then there will be totally $100 \cdot M$ comparisons between the consensus clusterings and the base clusterings over 100 runs. We call it a *win* if the consensus clustering has a higher NMI score than a base clustering and call it a *loss* if the consensus clustering has a lower NMI score than a base clustering. Ties count as 1/2 win and 1/2 loss. The winning percentage is defined as the number of wins divided by the total number of comparisons. As shown in Table 4, the GP-MGLA method and the WEAC method (associated with AL) outperform most of the base clustering w.r.t. the best number of clusters on the benchmark datasets. We also compare the consensus clusterings against the base clusterings w.r.t. the same number of clusters, which means for each comparison the number of clusters of the consensus clustering are set to the same number as the base clustering. As shown in Table 5, GP-MGLA and WEAC outperform about two thirds of the base clusterings w.r.t. the same number of clusters on the benchmark datasets.

7.4.2. Comparison with Other Clustering Ensemble Methods

We compare the proposed WEAC and GP-MGLA methods against five different clustering ensemble methods, namely, the hybrid bipartite graph formulation (HBGF) [14], the weighted consensus clustering (WCC) [31], the evidence accumulation clustering (EAC) [15], the ensemble clustering by matrix completion (ECMC) [24], the SimRank similarity based method (SRS) [19], and the weighted connected-triple method (WCT) [23]. Since the ECMC method and the WCC method is very time-consuming (see Fig. 4), it is almost infeasible to run ECMC and WCC for 100 times on the large datasets as the *Pen Digits* and *Letters* datasets, which contain 10,992 and 20,000 instances respectively. Therefore, the ECMC and WCC methods are performed on the benchmark datasets except the *Pen Digits* and *Letters* datasets. And the other baseline methods are performed on all the benchmark datasets.

The EAC, ECMC, SRS, and WCT methods are four pair-wise similarity based methods, each leading to three sub-methods by utilizing three different agglomerative clustering methods, namely, AL, CL, and SL. Then we have 14 baseline methods, that is, HBGF, WCC, EAC-AL, EAC-CL, EAC-SL, ECMC-AL, ECMC-CL, ECMC-SL, SRS-AL, SRS-CL, SRS-SL, WCT-AL, WCT-CL, and WCT-SL. The average performance over 100 runs of the proposed methods and the 14 baseline methods for each dataset is summarized in Table 6 and 7. For each test method, the number of clusters k for the consensus clustering is set to two values respectively, that is, best- k and true- k . The best- k is the number of clusters that leads to the optimal performance for a method on the dataset. The true- k is the number of true classes in the dataset. As shown in Table 6 and 7, the performance of the WEAC-AL method is better and more stable than the other pair-wise similarity based methods. The WEAC-AL method achieves the best NMI scores for the *Seeds* dataset and nearly best NMI scores for the *Iris*, *Image Segmentation*, *Yeast*, and *Wine* datasets. Among the test methods, the GP-MGLA method produces overall the best and most stable clustering results on the benchmark datasets.

7.4.3. Dealing with Ill Clusterings

In order to evaluate the robustness of our methods to ill clusterings, we add a certain ratio of heavily imbalanced clusterings into the base clustering pool. For example, adding 20% of ill clusterings into the pool means replacing 20% of base clusterings in the pool with heavily imbalanced clusterings. To

Table 6: Average performance in terms of NMI over 100 runs by different clustering ensemble methods (The two highest scores in each column are highlighted in bold.)

| Method | <i>Breast Cancer</i> | | <i>Iris</i> | | <i>Image Segmentation</i> | |
|---------|----------------------|----------------|----------------|----------------|---------------------------|----------------|
| | Best- <i>k</i> | True- <i>k</i> | Best- <i>k</i> | True- <i>k</i> | Best- <i>k</i> | True- <i>k</i> |
| GP-MGLA | 0.715 | 0.618 | 0.733 | 0.695 | 0.633 | 0.549 |
| WEAC+AL | 0.672 | 0.596 | 0.731 | 0.673 | 0.638 | 0.533 |
| WEAC+CL | 0.405 | 0.073 | 0.639 | 0.653 | 0.580 | 0.317 |
| WEAC+SL | 0.437 | 0.030 | 0.701 | 0.493 | 0.584 | 0.420 |
| HBGF | 0.695 | 0.648 | 0.707 | 0.640 | 0.631 | 0.491 |
| WCC | 0.621 | 0.459 | 0.694 | 0.539 | 0.621 | 0.527 |
| EAC+AL | 0.652 | 0.512 | 0.725 | 0.667 | 0.637 | 0.503 |
| EAC+CL | 0.421 | 0.058 | 0.637 | 0.497 | 0.582 | 0.323 |
| EAC+SL | 0.377 | 0.010 | 0.680 | 0.632 | 0.535 | 0.413 |
| ECMC+AL | 0.436 | 0.399 | 0.272 | 0.140 | 0.100 | 0.081 |
| ECMC+CL | 0.390 | 0.358 | 0.306 | 0.181 | 0.126 | 0.102 |
| ECMC+SL | 0.381 | 0.259 | 0.403 | 0.272 | 0.060 | 0.026 |
| SRS+AL | 0.650 | 0.519 | 0.726 | 0.676 | 0.642 | 0.513 |
| SRS+CL | 0.632 | 0.489 | 0.708 | 0.648 | 0.624 | 0.530 |
| SRS+SL | 0.544 | 0.029 | 0.706 | 0.661 | 0.619 | 0.411 |
| WCT+AL | 0.668 | 0.075 | 0.724 | 0.673 | 0.632 | 0.494 |
| WCT+CL | 0.621 | 0.110 | 0.698 | 0.644 | 0.615 | 0.492 |
| WCT+SL | 0.546 | 0.124 | 0.705 | 0.650 | 0.580 | 0.416 |

| Method | <i>Seeds</i> | | <i>Yeast</i> | | <i>Wine</i> | |
|---------|----------------|----------------|----------------|----------------|----------------|----------------|
| | Best- <i>k</i> | True- <i>k</i> | Best- <i>k</i> | True- <i>k</i> | Best- <i>k</i> | True- <i>k</i> |
| GP-MGLA | 0.595 | 0.514 | 0.254 | 0.167 | 0.797 | 0.717 |
| WEAC+AL | 0.604 | 0.517 | 0.256 | 0.147 | 0.767 | 0.664 |
| WEAC+CL | 0.449 | 0.197 | 0.212 | 0.093 | 0.540 | 0.177 |
| WEAC+SL | 0.517 | 0.317 | 0.210 | 0.046 | 0.614 | 0.235 |
| HBGF | 0.587 | 0.493 | 0.256 | 0.181 | 0.781 | 0.647 |
| WCC | 0.567 | 0.439 | 0.245 | 0.208 | 0.701 | 0.581 |
| EAC+AL | 0.582 | 0.399 | 0.256 | 0.109 | 0.733 | 0.444 |
| EAC+CL | 0.467 | 0.206 | 0.218 | 0.065 | 0.547 | 0.168 |
| EAC+SL | 0.502 | 0.244 | 0.173 | 0.034 | 0.580 | 0.104 |
| ECMC+AL | 0.233 | 0.126 | 0.073 | 0.021 | 0.270 | 0.154 |
| ECMC+CL | 0.238 | 0.146 | 0.074 | 0.022 | 0.266 | 0.159 |
| ECMC+SL | 0.216 | 0.064 | 0.073 | 0.017 | 0.253 | 0.078 |
| SRS+AL | 0.584 | 0.438 | 0.256 | 0.122 | 0.733 | 0.254 |
| SRS+CL | 0.547 | 0.356 | 0.229 | 0.116 | 0.658 | 0.407 |
| SRS+SL | 0.560 | 0.344 | 0.204 | 0.034 | 0.668 | 0.001 |
| WCT+AL | 0.573 | 0.403 | 0.268 | 0.120 | 0.730 | 0.478 |
| WCT+CL | 0.555 | 0.326 | 0.230 | 0.095 | 0.672 | 0.396 |
| WCT+SL | 0.528 | 0.289 | 0.221 | 0.035 | 0.639 | 0.184 |

Table 7: Average performance in terms of NMI over 100 runs by different clustering ensemble methods (The two highest scores in each column are highlighted in bold.)

| Method | <i>Pen Digits</i> | | <i>Letters</i> | |
|---------|-------------------|--------------|----------------|--------------|
| | Best- k | True- k | Best- k | True- k |
| GP-MGLA | 0.791 | 0.725 | 0.452 | 0.378 |
| WEAC+AL | 0.760 | 0.667 | 0.440 | 0.345 |
| WEAC+CL | 0.557 | 0.243 | 0.423 | 0.160 |
| WEAC+SL | 0.597 | 0.167 | 0.370 | 0.092 |
| HBGF | 0.781 | 0.663 | 0.445 | 0.364 |
| EAC+AL | 0.745 | 0.588 | 0.425 | 0.290 |
| EAC+CL | 0.584 | 0.244 | 0.322 | 0.139 |
| EAC+SL | 0.445 | 0.085 | 0.102 | 0.061 |
| SRS+AL | 0.750 | 0.611 | 0.424 | 0.300 |
| SRS+CL | 0.684 | 0.485 | 0.391 | 0.280 |
| SRS+SL | 0.706 | 0.127 | 0.070 | 0.032 |
| WCT+AL | 0.765 | 0.584 | 0.442 | 0.333 |
| WCT+CL | 0.661 | 0.392 | 0.414 | 0.278 |
| WCT+SL | 0.718 | 0.136 | 0.121 | 0.065 |

produce the heavily imbalanced clusterings, we firstly partition the dataset into k clusters via k -means where k is randomly chosen in the interval of $[\sqrt{n}, 2\sqrt{n}]$. Then we merge a proportion ρ of clusters into one, i.e., $\rho \cdot k$ randomly chosen clusters will be merged into one cluster in the clustering. In our experiments, the values of ρ are randomly selected in the interval of $(0.7, 0.99)$, which lead to heavily imbalanced clusterings. Different ratio of ill base clusterings are added to the pool and then we conduct experiments on the ensemble of randomly chosen base clusterings from the pool.

For each ratio of ill base clusterings, we run each of the clustering ensemble methods for 100 times and the performance is summarized in Fig. 3. The average-link is used for each of the pair-wise similarity based methods, namely, WEAC, EAC, ECMC, SRS, and WCT. As can be seen in Fig. 3, the proposed WEAC method yields much better performance than the EAC method on the benchmark datasets. On the whole, the proposed GP-MGLA method yields much better and more robust performance than the other clustering ensemble methods with different ratio of ill base clusterings added.

7.5. Computational Complexity

The computation of the NMI measure between two partitions takes $O(n^2)$ time, where n is the number of instances in the dataset. The computation of the NCAI measure takes $O(M^2n^2)$ time, where M is the number of base

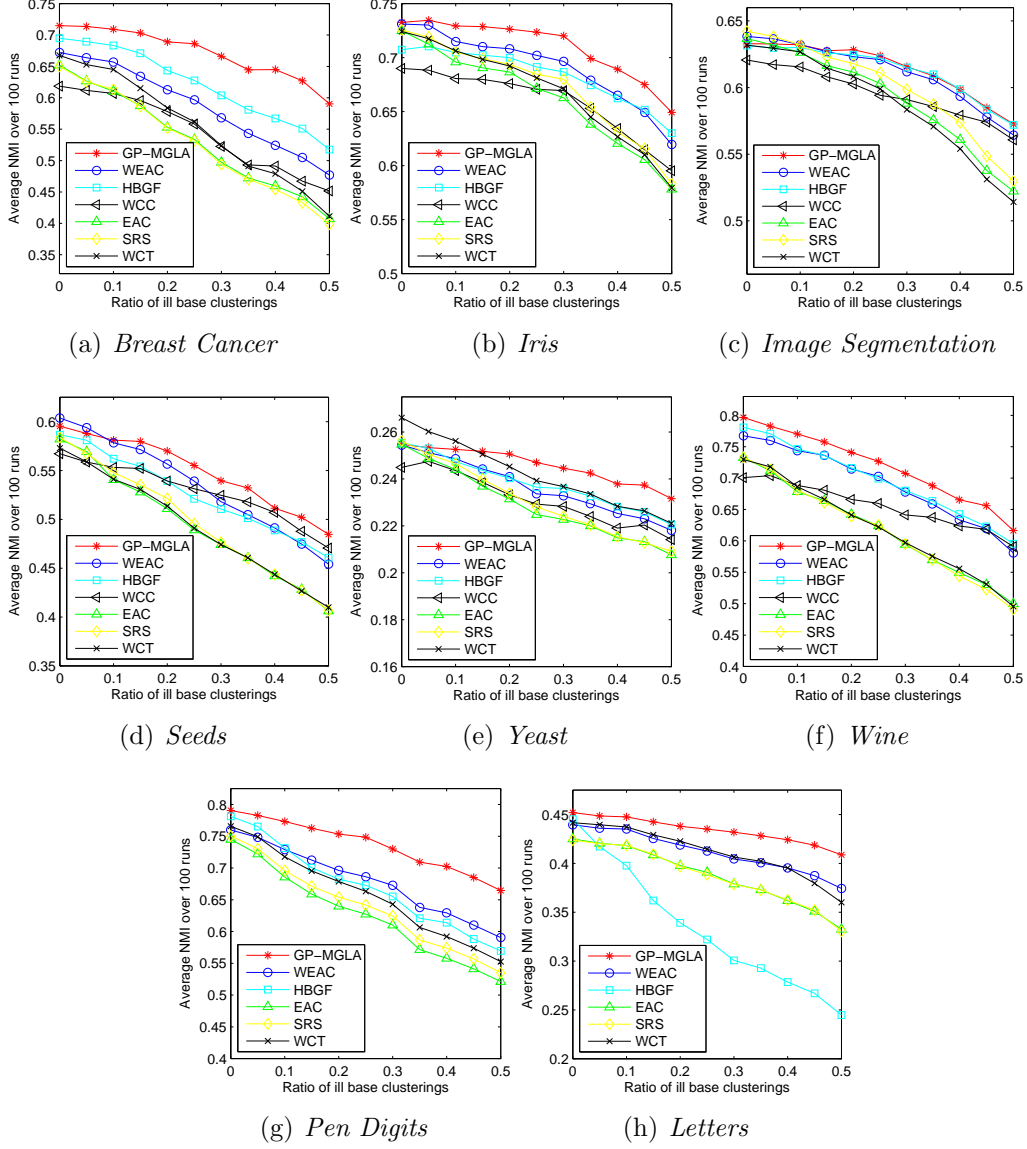


Figure 3: The performance by the proposed methods and the baseline methods over different ratio of ill base clusterings.

clusterings in the ensemble. The computation of the SACT similarity is $O(M^2n^2 + ln_c^2 + nn_c)$, where n_c is the number of clusters in the ensemble and l is the average number of neighbors connecting to a cluster. As the con-

ventional EAC method [15] is $O(Mn^2)$, the time complexity of the proposed WEAC method (associated with average-link) is $O(M^2n^2)$. The Tcut algorithm for bipartite graph partitioning is $O(kdn + kn_c^2)$, where k is the number of clusters in the final consensus clustering and d is the average number of links connecting to a node in the graph. Then we have the time complexity of the proposed GP-MGLA method as $O(M^2n^2 + (l + k)n_c^2 + nn_c + kdn)$.

The proposed methods and the baseline methods are applied to the *Letters* dataset to test the execution time w.r.t. varying data sizes. The time performance of these test methods with varying data sizes is illustrated in Fig. 4. To process the entire *Letters* dataset with 20,000 instances, the time costs (in seconds) of WEAC and GP-MGLA are 82.94 and 5.64 respectively, whereas the time costs (in seconds) of HBGF, EAC, and WCT are 2.51, 81.91, and 138.43 respectively. In the proposed methods, it takes 2.01 seconds to compute the NCAI for the data size of 20,000. Each of the five pair-wise similarity based methods, namely, WEAC, EAC, ECMC, SRS, and WCT, is associated with average-link. As shown in Fig. 4, the ECMC method and the WCC method are the two slowest methods. And the SRS method is the third slowest. The GP-MGLA is slower than WEAC, EAC, and WCT when the data size is below 4,000. However, the GP-MGLA shows an advantage in execution time as the data size grows beyond 5,000. The proposed GP-MGLA method and the HBGF method are the two fastest methods when the data size is greater than 5,000, mainly due to their efficient graph partitioning algorithms.

8. Conclusions

In this paper, we address the clustering ensemble problem using crowd agreement estimation and multi-granularity link analysis. With the clustering ensemble viewed as a crowd, we assess reliability of the individuals inside it by exploiting the so-called wisdom of the crowd. The normalized crowd agreement index is proposed for evaluating the quality of base clusterings in an unsupervised manner. The source aware connected triple similarity is introduced for constructing the link between two clusters with their common neighbors and source reliability taken into consideration. To achieve the final consensus clustering, two novel consensus functions are further presented, termed weighted evidence accumulation clustering (WEAC) and graph partitioning with multi-granularity link analysis (GP-MGLA) respectively. The

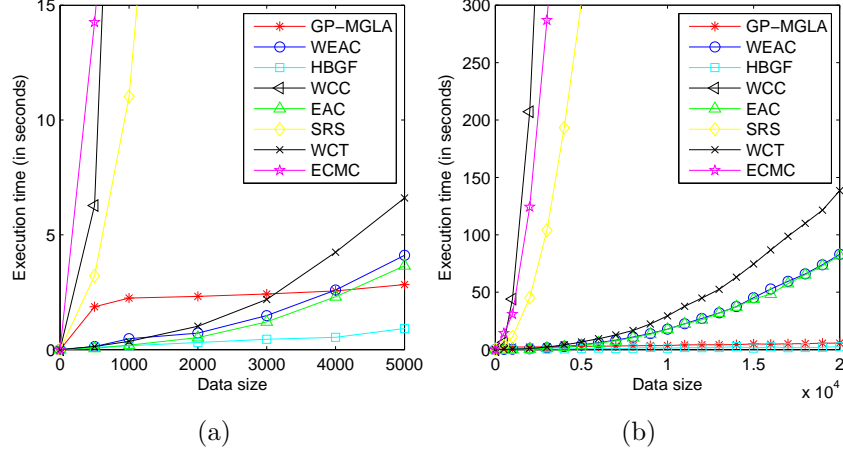


Figure 4: Execution time of different clustering ensemble approaches as the data size varies (a) from 0 to 5,000 and (b) from 0 to 20,000.

experiments conducted on eight real-world datasets show the effectiveness and robustness of the proposed clustering ensemble methods.

Acknowledgment

The authors would like to thank the anonymous reviewers for their insightful comments and suggestions which helped enhance this paper significantly. This work was supported by NSFC (61173084 and 61128009), National Science & Technology Pillar Program (No. 2012BAK16B06) and the Research Training Program of SMIE of Sun Yat-sen University.

References

- [1] L. Xu, A. Krzyzak, E. Oja, Rival penalized competitive learning for clustering analysis, RBF net, and curve detection, *IEEE Transactions on Neural Networks* 4 (4) (1993) 636–649.
- [2] J. Li, S. Ray, B. G. Lindsay, A nonparametric statistical approach to clustering via mode identification, *Journal of Machine Learning Research* 8 (2007) 1687–1723.
- [3] M.-L. Zhang, Z.-H. Zhou, Multi-instance clustering with applications to multi-instance prediction, *Applied Intelligence* 31 (1) (2009) 47–68.

- [4] F. Zhao, L. Jiao, H. Liu, X. Gao, M. Gong, Spectral clustering with eigenvector selection based on entropy ranking, *Neurocomputing* 73 (10-12) (2010) 1704–1717.
- [5] C.-D. Wang, J.-H. Lai, Energy based competitive learning, *Neurocomputing* 74 (12-13) (2011) 2265–2275.
- [6] M. Li, X. C. Lian, J. T. Kwok, B. L. Lu, Time and space efficient spectral clustering via column sampling, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR’11)*, 2011.
- [7] C.-D. Wang, J.-H. Lai, D. Huang, Incremental support vector clustering, in: *Proceedings of the IEEE International Conference on Data Mining Workshops (ICDMW’11)*, 2011.
- [8] C.-D. Wang, J.-H. Lai, C. Y. Suen, J.-Y. Zhu, Multi-exemplar affinity propagation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35 (9) (2013) 2223–2237.
- [9] C.-D. Wang, J.-H. Lai, D. Huang, W.-S. Zheng, SVStream: A support vector based algorithm for clustering data streams, *IEEE Transactions on Knowledge and Data Engineering* 25 (6) (2013) 1410–1424.
- [10] C.-D. Wang, J.-H. Lai, Position regularized support vector domain description, *Pattern Recognition* 46 (3) (2013) 875–884.
- [11] A. K. Jain, Data clustering: 50 years beyond k -means, *Pattern Recognition Letters* 31 (8) (2010) 651–666.
- [12] S. Vega-Pons, J. Ruiz-Shulcloper, A survey of clustering ensemble algorithms, *International Journal of Pattern Recognition and Artificial Intelligence* 25 (3) (2011) 337–372.
- [13] A. Strehl, J. Ghosh, Cluster ensembles: A knowledge reuse framework for combining multiple partitions, *Journal of Machine Learning Research* 3 (2003) 583–617.
- [14] X. Z. Fern, C. E. Brodley, Solving cluster ensemble problems by bipartite graph partitioning, in: *Proceedings of the International Conference on Machine Learning (ICML’04)*, 2004.

- [15] A. L. N. Fred, A. K. Jain, Combining multiple clusterings using evidence accumulation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27 (6) (2005) 835–850.
- [16] A. Topchy, A. K. Jain, W. Punch, Clustering ensembles: models of consensus and weak partitions, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27 (12) (2005) 1866–1881.
- [17] S. T. Hadjitodorov, L. I. Kuncheva, L. P. Todorova, Moderate diversity for better cluster ensembles, *Information Fusion* 7 (3) (2006) 264–275.
- [18] Y. Li, J. Yu, P. Hao, Z. Li, Clustering ensembles based on normalized edges, in: *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD’07)*, 2007.
- [19] N. Iam-On, T. Boongoen, S. Garrett, Refining pairwise similarity matrix for cluster ensemble problem with cluster relations, in: *Proceedings of the International Conference on Discovery Science (ICDS’08)*, 2008.
- [20] C. Domeniconi, M. Al-Razgan, Weighted cluster ensembles: Methods and analysis, *ACM Transactions on Knowledge Discovery from Data* 2 (4) (2009) 1–40.
- [21] X. Wang, C. Yang, J. Zhou, Clustering aggregation by probability accumulation, *Pattern Recognition* 42 (5) (2009) 668–675.
- [22] S. Mimaroglu, E. Erdil, Combining multiple clusterings using similarity graph, *Pattern Recognition* 44 (3) (2011) 694–703.
- [23] N. Iam-On, T. Boongoen, S. Garrett, C. Price, A link-based approach to the cluster ensemble problem, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33 (12) (2011) 2396–2409.
- [24] J. Yi, T. Yang, R. Jin, A. K. Jain, Robust ensemble clustering by matrix completion, in: *Proceedings of the IEEE International Conference on Data Mining (ICDM’12)*, 2012.
- [25] L. Franek, X. Jiang, Ensemble clustering by means of clustering embedding in vector spaces, *Pattern Recognition* 47 (2) (2014) 833–842.

- [26] D. Huang, J.-H. Lai, C.-D. Wang, Exploiting the wisdom of crowd: A multi-granularity approach to clustering ensemble, in: Proceedings of the International Conference on Intelligence Science and Big Data Engineering (IScIDE'13), 2013.
- [27] D. Cristofor, D. Simovici, Finding median partitions using information-theoretical-based genetic algorithms, *Journal of Universal Computer Science* 8 (2) (2002) 153–172.
- [28] E. Weiszfeld, F. Plastria, On the point for which the sum of the distances to n given points is minimum, *Annals of Operations Research* 167 (1) (2009) 7–41.
- [29] S. Vega-Pons, J. Correa-Morris, J. Ruiz-Shulcloper, Weighted partition consensus via kernels, *Pattern Recognition* 43 (8) (2010) 2712–2724.
- [30] S. Vega-Pons, J. Ruiz-Shulcloper, A. Guerra-Gandón, Weighted association based methods for the combination of heterogeneous partitions, *Pattern Recognition Letters* 32 (16) (2011) 2163–2170.
- [31] T. Li, C. Ding, Weighted consensus clustering, in: Proceedings of the SIAM International Conference on Data Mining (SDM'08), 2008.
- [32] X. Z. Fern, W. Lin, Cluster ensemble selection, *Statistical Analysis and Data Mining* 1 (3) (2008) 128–141.
- [33] S. Wu, T. W. S. Chow, Clustering of the self-organizing map using a clustering validity index based on inter-cluster and intra-cluster density, *Pattern Recognition* 37 (2) (2004) 175–188.
- [34] K. Faceli, M. C. P. de Souto, D. S. A. de Araújo, A. C. P. L. F. de Carvalho, Multi-objective clustering ensemble for gene expression data analysis, *Neurocomputing* 72 (2009) 2763–2774.
- [35] N. Li, L. J. Latecki, Clustering aggregation as maximum-weight independent set, in: Advances in Neural Information Processing Systems (NIPS'12), 2012.
- [36] J. Surowiecki, The wisdom of crowds: Why the many are smarter than the few and how collective wisdom shapes business, economies, societies and nations, Anchor Books, 2004.

- [37] M. Levandowsky, D. Winter, Distance between sets, *Nature* 234 (1971) 34–35.
- [38] Z. Li, X.-M. Wu, S.-F. Chang, Segmentation using superpixels: A bipartite graph partitioning approach, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'12)*, 2012.
- [39] K. Bache, M. Lichman, *UCI machine learning repository*, 2013.
- [40] D. Huang, J.-H. Lai, C.-D. Wang, Incremental support vector clustering with outlier detection, in: *Proceedings of the International Conference on Pattern Recognition (ICPR'12)*, 2012.